

Struktur- und Parameteridentifikation 2

Übung 2 - Nichtparametrische Identifikation im geschlossenen Regelkreis

In dieser Übung soll ein Modell für einen stark gestörten Prozess im geschlossenen Regelkreis nichtparametrisch identifiziert werden. Als **Einstieg in Simulink** sind die beiden Kurse 'MATLAB Onramp' und 'Simulink Onramp' auf der Homepage von Mathworks matlabacademy.mathworks.com/ empfehlenswert. Der Versuch in dieser Rechnerübung wird durch das Simulink Modell `Identversuch_stabil.mdl` simuliert. Für die **Identifikation** wird ein in Matlab erzeugtes **PRBS**-Signal genutzt, für die **Kreuzvalidierung** wird eine **Sägezahnfunktion** angesetzt. Nutzen Sie zur Bearbeitung das Skript `uebung2.m`.

1. Wiederholungsfragen

- Wann ist eine Identifikation im geschlossenen Regelkreis notwendig? Instabilität der Strecke, Qualitäts-Anforderungen
- Warum ist eine direkte Identifikation problematisch? Rückkopplung: Störung mit Streckeneingang korreliert über Feedback-Loop --> Bias in Parameter-Schätzung
- Welche Verfahren gibt es für die Identifikation im geschlossenen Regelkreis?
 - Klassisch: nichtparametrische Identifikation über spektrale Leistungsdichten; liefert nicht-parametrisches Modell
 - Neu: Prediction-Error-Methode (PEM); liefert parametrisches Modell; dafür 4 ÜF (M,L,G_S & K) bestimmen

2. Nichtparametrische Identifikation

Im Ordner 'Zur Vorbereitung für die 2 UEB spi 2' finden Sie das Simulink-Modell '`Identversuchstabil.mdl`'. Im m-file '`spi1_uebung3_LSG`' finden Sie das Skript, um das Modell geeignet anzuregen, um die Daten für die Nichtparametrische Identifikation sowie für die Kreuzvalidierung zu erhalten. Für die Wiederholung der Identifikation und um sich mit dem Modell im geschlossenen Regelkreis vertraut zu machen, wird empfohlen die Aufgabe 2 der UEB 3 von spi 1 durchzurechnen.

- x a) Regen Sie das System mit einem **PRBS** für die Identifikation bzw. Autovvalidierung und mit dem **Sägezahn** für die Kreuzvalidierung an, um die Daten u_I, y_I, w_I sowie u_K, y_K, w_K zu erzeugen. **Plotten** Sie die Daten. Nutzen Sie dafür das Skript 'Zur Vorbereitung für die 2 UEB spi 2'.
- x a2) Schätzen Sie mit `delayest` die Totzeit der Strecke ab
- b) Berechnen Sie die **Kreuzleistungsdichten** von der Führungsgröße und dem Ausgang sowie von der Stell- und Führungsgröße. Nutzen Sie dafür geeignete Befehle, die Sie aus der UEB 1 - SPI 1 kennengelernt haben. Ein weiteres Verfahren das **Welch**-Verfahren wurde im Vorlesungsskript Seite 98 vorgestellt und die genauen Matlab Befehle für die Berechnung der Kreuzspektraldichte auf Seite 101 präsentiert. Probieren Sie die Methoden aus. Die Kreuzkorrelation lässt sich in Matlab mit dem Befehl `xcorr(x,y)` berechnen.
- x c) Identifizieren Sie die nicht parametrische Übertragungsfunktion nach Gleichung (3.149) auf Seite 204 im Skript und überführen Sie es in ein parametrisches Modell mithilfe der Funktionen aus UEB 1 - SPI 2 (`myVectorFit`).
- x d) Als Ergebnis erhält man ein **Zustandsraummodell**. Vergleichen Sie Ihr Ergebnis mit dem wahren Modell aus Simulink. (Umrechnen mit `tf2ss` in Matlab).
- x e) bezogen auf b:
Die Kreuzleistungsdichte ist ein komplexer Vektor. Ist sie dennoch kommutativ? Ist also $\backslash\Phi_{wy} = \backslash\Phi_{yw}$?
- x f) Aktualisiere deine falschen Executive Summaries im blauen Notizbuch zur Berechnung der Periodogramme mit Matlab! --> Man muss nur einmal mit N normieren, nicht auch noch jeweils die Ergebnisse der Fourier-Trafos!
- x g) Versuche die beiden Welch-Methoden, sobald du IWA hast und die Signal Processing Toolbox ziehen kannst!
- h) notiere die Zusammenfassungen im blauen Heft!
Notiere ebenso die vier Verfahren aus dem Skript (SPI 1+2, S.107), mit denen G geschätzt wird!